

BAB III

LANDASAN TEORI

3.1 Bidan Praktek Swasta

Bidan Praktek Swasta (BPS), merupakan penyedia layanan kesehatan, yang memiliki kontribusi cukup besar dalam memberikan pelayanan, khususnya dalam meningkatkan kesejahteraan ibu dan anak. Supaya masyarakat pengguna jasa layanan bidan memperoleh akses pelayanan yang bermutu dari pelayanan bidan, perlu adanya regulasi pelayanan praktek bidan secara jelas, persiapan sebelum bidan melaksanakan pelayanan praktek, seperti perizinan, tempat, ruangan, peralatan praktek, dan kelengkapan administrasi semuanya harus sesuai dengan standar, diselenggarakan oleh lebih dari satu jenis tenaga kesehatan dan dipimpin oleh seorang tenaga medis.

BPS bertanggung jawab menyelenggarakan pembangunan kesehatan khususnya di bidang perbidanan dalam suatu wilayah kerja. BPS hanya bertanggung jawab untuk sebagian upaya pembangunan kesehatan. Secara nasional, standar wilayah kerja BPS adalah satu kecamatan. Tetapi apabila disatu kecamatan terdapat lebih dari satu BPS, maka tanggung jawab wilayah kerja dibagi antar BPS dengan memperhatikan keutuhan konsep wilayah (desa, kelurahan, rukun warga), dan masing-masing BPS tersebut secara operasional bertanggung jawab langsung kepada dinas kesehatan kabupaten/ kota (Sulastomo, 2007)).

Visi pembangunan kesehatan yang diselenggarakan oleh klinik BPS Farida Hadjri adalah sebagai tempat pelayanan Kesehatan Masyarakat dan

Kebidanan yang aman dan Profesional demi terwujudnya Indonesia sehat khususnya di Kecamatan Nyamplungan. Kecamatan sehat adalah gambaran masyarakat kecamatan masa depan yang ingin dicapai melalui pembangunan kesehatan, yakni masyarakat yang hidup didalam lingkungan dengan perilaku sehat, memiliki kemampuan untuk menjangkau pelayanan kesehatan yang bermutu secara adil dan merata serta memiliki derajat kesehatan yang setinggi-tingginya (Sulastomo, 2007).

Misi pembangunan kesehatan yang diselenggarakan BPS adalah mendukung tercapainya misi pembangunan kesehatan nasional, yaitu :

1. Menggerakkan pembangunan berwawasan kesehatan di wilayah kerjanya.
2. Mendorong kemandirian hidup sehat bagi keluarga dan masyarakat di wilayah kerjanya.
3. Memelihara dan meningkatkan mutu, pemerataan dan keterjangkauan pelayanan kesehatan yang diselenggarakan khususnya dalam bidang kebidanan.
4. Memelihara dan meningkatkan kesehatan per orangan, keluarga, dan masyarakat, serta lingkungannya (Direktorat Jaminan Pemeliharaan Kesehatan Masyarakat, 2003).

3.2 Pengelolaan Data

Pengelolaan adalah suatu sistem yang akan mengatur bentuk bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi. Pengelolaan data adalah serangkaian operasi informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan (Karissa, 2012). Sedangkan data adalah sebagai bahan

mentah dari informasi yang dirumuskan sebagai kelompok lambang-lambang tidak acak yang menunjukkan jumlah atau tindakan-tindakan (Karissa, 2012).

3.3 Sistem Informasi

Menurut Herlambang dan Tanuwijaya (2005:121), data adalah fakta-fakta atau kejadian-kejadian yang dapat berupa angka-angka atau kode-kode tertentu. Data masih belum mempunyai arti bagi penggunanya. Untuk dapat mempunyai arti data diolah sedemikian rupa sehingga dapat digunakan oleh penggunanya. Hasil pengolahan data inilah yang disebut sebagai informasi. Secara ringkas, Informasi adalah data yang telah diolah dan mempunyai arti bagi penggunanya. Sehingga sistem informasi dapat didefinisikan sebagai prosedur-prosedur yang digunakan untuk mengolah data sehingga dapat digunakan oleh penggunanya.

Informasi merupakan data yang telah diolah menjadi suatu bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau di masa yang akan datang. Sumber dari informasi adalah data, merupakan bentuk yang masih mentah dan belum dapat bercerita banyak, sehingga membutuhkan pengolahan lebih lanjut. Kualitas dari sistem informasi bergantung pada dua hal, yaitu:

1. Informasi harus akurat, dimana informasi tersebut harus bebas dari kesalahan.
2. Informasi tersebut harus relevan, supaya informasi tersebut dapat memberikan masukan bagi penerimanya.

Istilah sistem informasi juga sering di kacaukan dengan Sistem Informasi Manajemen (SIM). Kedua hal ini sebenarnya tidak sama. Sistem informasi

manajemen merupakan salah satu jenis sistem informasi, yang secara khusus ditujukan untuk menghasilkan informasi bagi pihak manajemen dan untuk pengambilan keputusan.

3.4 Sistem Flow

System flow adalah bagan yang menunjukkan arus pekerjaan secara menyeluruh dari suatu sistem dimana bagan ini menjelaskan urutan prosedur-prosedur yang ada di dalam sistem dan biasanya dalam membuat *system flow* ditentukan oleh fungsi yang melaksanakan dan yang bertanggung jawab (Kendall, et.al;2003:56).

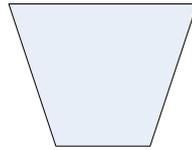
Terdapat berbagai macam bentuk simbol yang digunakan untuk merancang sebuah desain dari sistem, diantaranya adalah *terminator*, *manual operation*, *document*, *process*, *database*, *manual input*, *decision*, *off-line storage*, *on-page reference*, dan *off-page reference*.

Terminator merupakan bentuk simbol yang di gunakan sebagai tanda di mulainya jalan proses sistem ataupun tanda akhir dari sebuah pengerjaan suatu sistem. Simbol dari *terminator* dapat dilihat pada gambar 3.1.



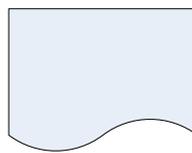
Gambar 3.1 *Terminator*

Manual operation digunakan untuk menggambarkan sebuah proses kerja yang dilakukan tanpa menggunakan komputer sebagai medianya (menggunakan proses manual). Simbol dari *manual operation* dapat dilihat pada gambar 3.2.



Gambar 3.2 *Manual Operation*

Document merupakan simbol dari dokumen yang berupa kertas laporan, surat-surat, memo, maupun arsip-arsip secara fisik. Simbol dari *document* dapat dilihat pada gambar 3.3.



Gambar 3.3 *Document*

Process adalah sebuah bentuk kerja sistem yang dilakukan secara terkomputerisasi. Simbol dari *process* dapat dilihat pada gambar 3.4.



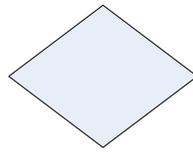
Gambar 3.4 *Process*

Database digunakan sebagai media penyimpanan data yang bersifat terkomputerisasi. Simbol dari *database* dapat di lihat pada gambar 3.5.



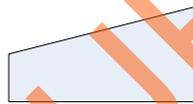
Gambar 3.5 *Database*

Decision merupakan operator logika yang digunakan sebagai penentu keputusan dari suatu permintaan atau proses dengan dua nilai, benar dan salah. Simbol dari *decision* dapat dilihat pada gambar 3.6.



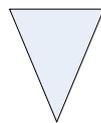
Gambar 3.6 *Decision*

Manual input digunakan untuk melakukan proses *input* kedalam database melalui keyboard. Simbol dari *manual input* dapat dilihat pada gambar 3.7.



Gambar 3.7 *Manual Input*

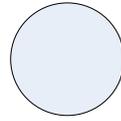
Off-line storage merupakan bentuk media penyimpanan yang berbeda dengan *database*, dimana media penyimpanan ini menyimpan dokumen secara manual atau lebih dikenal dengan nama arsip. Simbol dari *off-line storage* dapat dilihat pada gambar 3.8.



Gambar 3.8 *Off – line Storage*

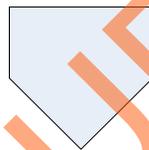
On-page reference digunakan sebagai simbol untuk menghubungkan bagan desain sebuah sistem apabila hubungan arus data yang ada terlalu jauh

dalam permasalahan letaknya. Simbol dari *on-page reference* dapat dilihat pada gambar 3.9.



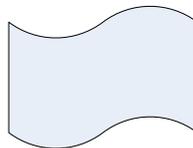
Gambar 3.9 *On – page reference*

Off-page reference memiliki sifat yang sedikit berbeda dengan *on-page reference*, karena simbol ini hanya digunakan apabila arus data yang ada dilanjutkan ke halaman yang berbeda. Simbol dari *off-page reference* dapat dilihat pada gambar 3.10.



Gambar 3.10 *Off – page reference*

Papertape merupakan sebuah simbol yang umumnya menggantikan bentuk penggambaran jenis pembayaran yang digunakan (misal : uang) dalam transaksi yang ada pada sistem yang di rancang. Simbol dari *paper tape* dapat dilihat pada gambar 3.11.



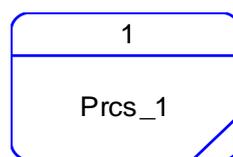
Gambar 3.11 *Paper Tape*

3.5 Data Flow Diagram (DFD)

Menurut Kristianto (2004), *Data Flow Diagram* (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data tersebut disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan, dan proses yang dikenakan pada data tersebut.

DFD merupakan suatu metode pengembangan sistem yang terstruktur (*structure analysis and design*). Penggunaan notasi dalam DFD sangat membantu untuk memahami suatu sistem pada semua tingkat kompleksitas. Pada tahap analisis, penggunaan notasi ini dapat membantu dalam berkomunikasi dengan pemakai sistem untuk memahami sistem secara logika.

Didalam DFD, terdapat empat simbol yang digunakan yaitu *process*, *external entity*, *data store*, dan *data flow*. Simbol *process* digunakan untuk melakukan suatu perubahan berdasarkan data yang diinputkan dan menghasilkan data dari perubahan tersebut. Simbol dari *process* dapat dilihat pada gambar 3.12.



Gambar 3.12 *Process*

Pada bentuk gambar *process*, bagian atas berisi nomor untuk identitas proses. Suatu proses dengan nomor 0 (nol atau kosong) menandakan bahwa proses tersebut adalah *context diagram*. Diagram ini merupakan level tertinggi dari DFD yang menggambarkan hubungan sistem dengan lingkungan luarnya. Pembuatan *context diagram* dapat dilakukan dengan terlebih dahulu menentukan

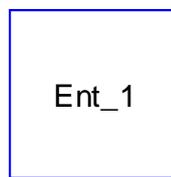
nama sistemnya, menentukan batasan dari sistem, dan menentukan *terminator* yang diterima atau diberikan daripada sistem untuk kemudian dilakukan penggambaran.

Nomor 1, 2, 3, dan seterusnya menandakan bahwa proses tersebut diartikan sebagai proses level-0 (nol) yang merupakan hasil turunan atau *decompose* dari proses *context diagram*. Proses level-0 membahas sistem secara lebih mendetil, baik dipandang dari segi kegiatan dari sebuah bagian, alur data yang ada, maupun *database* yang digunakan di dalamnya. Pembuatannya dapat dilakukan dengan cara menentukan proses utama yang ada dalam sistem, menentukan alur data yang diterima dan diberikan masing-masing proses dari pada sistem sambil memperhatikan konsep keseimbangan (alur data yang masuk atau keluar dari suatu level harus sama dengan alur data yang masuk dan keluar pada level berikutnya), memunculkan *data store* sebagai sumber maupun tujuan data (optional), menggambarkan diagram level-0, menghindari perpotongan arus data, dan melakukan pemberian nomor pada proses utama (nomor tidak menunjukkan urutan proses).

Nomor 1.1, 1.2, 2.1, 2.2, dan seterusnya merupakan sebuah proses turunan atau *decompose* dari proses level-0 yang disebut sebagai proses level-1 (satu). Proses level-1 menggambarkan detil kerja dari sebuah bagian dalam sebuah sistem. Penggambarannya dilakukan dengan cara menentukan proses yang lebih kecil (sub-proses) dari proses utama yang ada di level-0, menentukan apa yang diterima atau diberikan masing-masing sub-proses daripada sistem dan tetap memperhatikan konsep keseimbangan, memunculkan *data store* sebagai sumber maupun tujuan alur data (optional), menggambar DFD level-1, dan berusaha

untuk menghindari perpotongan arus data. Hasil turunan akhir disebut sebagai *the lowest level*, di mana hasil akhir ini tergantung dari kompleksitas sistem yang ada.

External entity disimbolkan dengan bentuk persegi yang digunakan untuk menggambarkan pelaku-pelaku sistem yang terkait, dapat berupa orang-orang, organisasi maupun instansi. *External entity* dapat memberikan masukan kepada *process* dan mendapatkan keluaran dari *process*. Simbol *external entity* dapat dilihat pada gambar 3.13.



Gambar 3.13 *External Entity*

Data store digunakan sebagai media penyimpanan suatu data yang dapat berupa *file* atau *database*, arsip atau catatan manual, lemari *file*, dan tabel-tabel dalam *database*. Penamaan *data store* harus sesuai dengan bentuk data yang tersimpan pada *data store* tersebut, misalnya tabel pelanggan, tabel detail penjualan, tabel detail pembelian, dan lain-lain. Simbol *data store* dapat dilihat pada gambar 3.14.



Gambar 3.14 *Data Store*

Data flow merupakan penghubung antar *external entity* dengan *process* dan *process* dengan *data store*. *Data flow* menunjukkan aliran data dari satu titik ke titik lainnya dengan tanda anak panah mengarah ke tujuan data. Penamaan *data*

flow harus menggunakan kata benda, karena didalam *data flow* mengandung sekumpulan data. Simbol *data flow* dapat dilihat pada gambar 3.15



Gambar 3.15 *Data Flow*

3.6 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah gambaran pada sistem dimana di dalamnya terdapat hubungan antara *entity* beserta relasinya. *Entity* merupakan sesuatu yang ada dan terdefiniskan di dalam suatu organisasi, dapat abstrak dan nyata. Untuk setiap *entity* biasanya mempunyai *atribute* yang merupakan ciri *entity* tersebut. Relasi adalah hubungan antar *entity* yang berfungsi sebagai hubungan yang mewujudkan pemetaan antar *entity*. (Marlinda, 2004)

Menurut Marlinda (2004), *Atribute* adalah kolom di sebuah relasi. Macam-macam *atribute* yaitu:

a. *Simple Atribute*

Atribute ini merupakan *atribute* yang unik dan tidak dimiliki oleh *atribute* lainnya, misalnya *entity* mahasiswa yang *atribute*-nya NIM.

b. *Composite Atribute*

Composite atribute adalah *atribute* yang memiliki dua nilai harga, misalnya nama besar (nama keluarga) dan nama kecil (nama asli).

c. *Single Value Atribute*

Atribute yang hanya memiliki satu nilai harga, misalnya *entity* mahasiswa dengan *atribute*-nya Umur (tanggal lahir).

d. *Multi Value Attribute*

Multi value attribute adalah *attribute* yang banyak memiliki nilai harga, misalnya *entity* mahasiswa dengan *attribute*-nya pendidikan (SD, SMP, SMA).

e. *Null Value Attribute*

Null value attribute adalah *attribute* yang tidak memiliki nilai harga, misalnya *entity* mahasiswa dengan *attribute* hobi (tanpa memiliki hobi).

Entity Relationship Diagram ini diperlukan agar dapat menggambarkan hubungan antar *entity* dengan jelas, dapat menggambarkan batasan jumlah *entity* dan partisipasi antar *entity*, mudah dimengerti pemakai dan mudah disajikan oleh perancang *database*. Untuk itu *Entity Relationship Diagram* dibagi menjadi dua jenis model, yaitu:

a. *Conceptual Data Model (CDM)*

Conceptual Data Model (CDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara konseptual.

b. *Physical Data Model (PDM)*

Physical Data Model (PDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara fisikal.

3.7 Program Penunjang

Untuk membuat sistem Informasi Pengelolaan Data Pasien di BPS Farida Hadjri, dibutuhkan beberapa perangkat lunak untuk memudahkan perancangan design maupun sistem. Perangkat lunak tersebut antara lain:

3.7.1 Power Designer

Power designer merupakan suatu *tool* berupa *software* untuk mendesain sistem dan rancangan *Entity Relationship Diagram* (ERD) yang dikembangkan oleh *Sybase Inc.* Ada dua model data, yaitu : *Entity Relationship Diagram* (ERD) dan model relasional. Keduanya menyediakan cara untuk mendeskripsikan perancangan basis data pada peringkat logika.

Model ERD atau *Conceptual Data Model* (CDM) : model yang di buat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas (*entity*) serta hubungan (*relationship*) antara entitas-entitas itu.

Model Relasional atau *Physical Data Model* (PDM) : model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap tabel mempunyai sejumlah kolom dimana setiap kolom memiliki nama yang unik.

3.7.2 Visual Basic .NET

Visual Basic sudah umum digunakan untuk membuat aplikasi software. Microsoft Visual Basic .NET adalah sebuah alat untuk mengembangkan dan membangun aplikasi yang bergerak di atas sistem .NET Framework, dengan menggunakan bahasa BASIC (Yuswanto,2008). Dengan menggunakan alat ini, para programmer dapat membangun aplikasi *Windows Forms*, Aplikasi web berbasis ASP.NET, dan juga aplikasi *command-line*. Alat ini dapat diperoleh secara terpisah dari beberapa produk lainnya (seperti Microsoft Visual C++, Visual C#, atau Visual J#), atau juga dapat diperoleh secara terpadu dalam Microsoft Visual Studio .NET. Bahasa Visual Basic .NET sendiri menganut

paradigma bahasa pemrograman berorientasi objek yang dapat dilihat sebagai evolusi dari Microsoft Visual Basic versi sebelumnya yang diimplementasikan di atas .NET Framework. Peluncurannya mengundang kontroversi, mengingat banyak sekali perubahan yang dilakukan oleh Microsoft, dan versi baru ini tidak kompatibel dengan versi terdahulu.

3.7.3 SQL Server

SQL Server sudah banyak tersedia atau sudah umum digunakan dan lebih mudah penggunaannya. Microsoft .NET Framework (dibaca Microsoft Dot Net Framework) adalah sebuah komponen yang dapat ditambahkan ke sistem operasi Microsoft Windows atau telah terintegrasi ke dalam Windows (mulai dari Windows Server 2003 dan versi-versi Windows terbaru). Kerangka kerja ini menyediakan sejumlah besar solusi-solusi program untuk memenuhi kebutuhan-kebutuhan umum suatu program baru dan mengatur eksekusi program-program yang ditulis secara khusus untuk *framework* ini. .NET Framework adalah kunci penawaran utama dari Microsoft, dan dimaksudkan untuk digunakan oleh sebagian besar aplikasi-aplikasi baru yang dibuat untuk platform Windows.

3.7.4 Crystal Report

Merupakan *software* yang digunakan untuk pembuatan laporan. Dengan cara mengoneksi nama tabel yang akan dibuatkan laporannya. Setelah tampilan data ada maka klik dan *drag* semua *field* yang ada sesuai dengan tampilan yang diinginkan. Biasanya *crystal report* adalah komponen dari Visual Basic .NET.

3.8 Analisa Dan Perancangan Sistem

Analisis sistem didefinisikan sebagai uraian dari sistem informasi yang besar dan utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya. Langkah-langkah dasar dalam melakukan analisa sistem :

1. Identifikasi masalah
2. Memahami kerja dari sistem
3. Menganalisa sistem
4. Membuat laporan hasil analisis

Menurut Kendall dan Kendall (2003), analisis dan perancangan sistem dipergunakan untuk menganalisis, merancang, dan mengimplementasikan peningkatan-peningkatan fungsi bisnis yang dapat dicapai melalui penggunaan sistem informasi terkomputerisasi. Analisis sistem dilakukan dengan tujuan untuk dapat mengidentifikasi dan mengevaluasi permasalahan yang terjadi dan kebutuhan yang diharapkan, sehingga dapat diusulkan perbaikannya.

Perancangan sistem merupakan penguraian suatu sistem informasi yang utuh ke dalam bagian komputerisasi yang dimaksud, mengidentifikasi dan mengevaluasi permasalahan, menentukan kriteria, menghitung konsistensi terhadap kriteria yang ada, serta mendapatkan hasil atau tujuan dari masalah tersebut serta mengimplementasikan seluruh kebutuhan operasional dalam membangun aplikasi.